近似動的計画法による多品種サプライチェーンの最適制御 (確率的環境下での意思決定解析)

AUTHOR(S):

大野, 勝久

CITATION:

大野, 勝久. 近似動的計画法による多品種サプライチェーンの最適制御 (確率的環境下での意思決定解析). 数理解析研究所講究録 2013, 1864: 100-109

ISSUE DATE:

2013-11

URL:

http://hdl.handle.net/2433/195348

数理解析研究所講究録
第1864巻 2013年 100-109

近似動的計画法による多品種サプライチェーンの最適制御

愛知工業大学・経営学部　大野 勝久

Katsuhisa Ohno
Faculty of Business Administration, Aichi Institute of Technology

1 はじめに

サプライチェーン（supply chain、SC）は、最終消費者の需要変動をはじめ、生産における原材料・部品の供給遅延、設備故障、物流における交通渋滞、災害の発生等々、様々な不確実性に直面している。このような不確実性のもとでは、将来の時点における発注・生産・発送量を、現在の情報だけに基づいて予め決定することは合理的ではない。むしろ、将来のその時点における SC の状態を観測し、それに応じた発注・生産・発送量を決めるべきであり、この決め方を**発注・生産・発送政策**と呼ぶことにする。一般に、下流工程の在庫・仕掛品等の状態変化に同期して、上流工程へ発注・生産指示が出される方式を**プル方式**と呼んでいるが、状態に依存して決まる決定であり、明らかに発注・生産・発送政策に属する。これに対して、現在の情報だけに基づいて、SC 全般の将来にわたる全体最適を目指した発注･生産･発送計画を作成する先進的計画システム（advanced planning system、**APS**）は**プッシュ方式**であり、不確実性を何ら考慮することなく、最適性を喧伝している。しかし、「サプライチェーンハンドブック」12 章［1］では、APS の中で最善とされる線形計画法（linear programming, **LP**）に基づく最適化とプル方式の**基点在庫(base stock)方式**を、7 種類の部品を用いて 4 品種の最終製品を生産する生産システムを対象に、数値的に比較している。その結果、当然のことではあるが、基点在庫政策が LP に基づく APS を大幅に上回ることを示している。しかし［**2,3**］等に示すように、基点在庫政策は本研究の目的である**準最適発注・生産・発送政策**と比較すれば、ライン構成により異なるが、パラメータを最適設定した最高性能においても、平均費用を 3.2%～15.6%以上増加させており、決して満足できる政策ではない。**かんばん(kanban)方式**に始まるプル方式は、これまでに代表的なものとして、基点在庫方式、**CONWIP**、**ハイブリッド(hybrid)方式**、**拡張かんばん(extended kanban)方式**が提案されている。

不確実環境下におけるSCの最適制御問題は、基本的に**マルコフ決定過程**(Markov decision process, **MDP**)の問題であり、特に単位期間当りの平均費用を最小化する問題は時間平均費用問題（undiscounted Markov decision process, **UMDP**)として定式化される。しかしながら、MDPはR. Bellman**[4]**によって1950年代に提案された動的計画法(dynamic programming, **DP**)の1分野であり、DPの持つ弱点である「**次元の呪い**（curse of dimensionality）」を引き継いでいる。すなわち、次元(工程、品種)が増えるにつれ状態数が指数的に増大し、実用時間内に解くことが不可能となる欠点である。実際1987年**[5]**において、3779状態を持つ3工程生産ラインの最適制御問題を、MDPの厳密アルゴリズムである修正政策反復法（modified policy iteration method, **MPIM**）を用いて解いているが、当時4工程以上への拡張は不可能なことを痛感したものである。ところが、1996年 Bertsekas and Tsutsuklis**[6]** は "Neuro-

dynamic Programming"を著し、次元の呪いを克服する、**強化学習**を含む様々な試みをまとめて**ニューロDP**と呼んだ。その後、遷移先状態数、可能決定数を**第2、第3の次元の呪い**と呼び、次元の呪いを克服する取り組みは"approximate dynamic programming"(**近似DP**) **[7,8]**と呼ばれるようになっている。

当時筆者は、ジャストインタイム（**JIT**）生産システムのIT活用による進化を目指して、引き取りかんばんと生産指示かんばんに代わりUMDPによる最適発注・生産政策を用いる制御を研究していた。そこで早速、単一工程JIT生産ラインの144状態を持つUMDPを既存の強化学習等の**近似DP**アルゴリズム［**9-11**］で解いてみた。ところが、［**2,3,12,14,15**］に示すように、かんばん枚数を最適化したかんばん政策の10倍以上費用がかかる政策しか得られなかった。そこでまず、MPIMにシミュレーションを併用する近似DPアルゴリズム**SBMPIM**（simulation-based modified policy iteration method）を提案し、同じ単一工程JIT生産ラインへ適用してMPIMと一致する最適政策が得られることを確認した。次いで、12,150状態を持つ2工程JIT生産システムへ適用し、最適化したかんばん方式より平均費用を8%～20%以上低減できることを示した**[12,15]**。さらに、SBMPIMアルゴリズムを143万状態を持つ3工程JIT生産・物流システムへ適用できるように改良し、拡張かんばん方式を除くプル方式をシミュレーションにより最適化した最高性能と準最適政策の性能を比較し、各プル方式が準最適政策にどれだけ近いかを明らかにしている**[13,14]**。その後、SBMPIMの改良を進めるとともに、プル方式のパラメータを最適設定する**SBOS**(simulation-based optimal setting)アルゴリズム**[2,3,16]**を開発して拡張かんばん方式を最適化し、すべてのプル方式と準最適政策を比較した結果を**[2,3]**に示している。その後、SBMPIMの初期政策として最適化したかんばん政策を採用し、それをシミュレーションしてMPIMの値近似ルーチンと結合することで初期政策の相対値を推定し、SBMPIMアルゴリズムの計算時間を１/200に短縮し、必要メモリーも大幅に減らすことに成功している。この新しい近似DPアルゴリズムを**SBMPI**（simulation-based modified policy iteration）アルゴリズムと呼び、第1番目の次元の呪いである状態空間を縮約し、第3番目の次元の呪いである決定空間を近傍化することで達成している**[17,18]**。

これまで研究を進めてきたSCは、SBMPIMアルゴリズムの性能による制約から、単一品種直列型SCに限定せざるを得なかった。しかし、SBMPIアルゴリズムが性能を飛躍的に向上させたので、不確実環境下における多品種SCの最適制御問題を取り扱えることとなった。

本論文では、次の**3階層**

１）最終消費者に複数の製品を販売する複数の小売店からなる第1階層

２）それら小売店へ複数の製品を供給する複数の配送センターからなる第2階層

３）それら配送センターへ複数の製品を供給する工場（第3階層）

からなる3階層多品種SCを取り扱う。このとき直列型では問題にならなかったが、複数の下流拠点への発送量を手持ちの製品在庫から割当なければならない。すなわち、単位期間当りの平均費用を最小化する最適な発注・生産・発送政策を求めることが目的となる。以下、この問題をUMDPとして定式化し、近似DPアルゴリズムSBMPIを用いて3階層多品種SCの準最適発注・生産・発送政策を決定する。

2 多品種サプライチェーンの最適制御

図１に示される、サプライヤーからp種類の部品の供給を受け、K品種の製品を生産する工場とそれら製品を取り扱う

L箇所の配送センターDj, j=1,...,L,およびM店の小売店Si, i=1,...,Mからなる3階層SCを考える。工場、Djおよび各小売店Siへの発注、生産指示、発送は各期首に行われる。サプライヤーは、十分な供給能力を持ち、輸送時間T_f (≥ 0)を含む一定の納入リードタイムL_f ($>T_f$)期後に受注した原材料を工場へ納入する。工場は輸送時間T_{Dj}(>0)を含む一定の納入リードタイムL_{Dj} ($\geq T_{Dj}$)期後に受注した製品をDjへ納入する。さらに、Djは小売店Si (i=1,...,M) への輸送時間Ts_i(≥ 0)を含む一定の納入リードタイムLs_i ($\geq Ts_i$)期後に受注した製品を小売店Siへ納入する。ただし、必要な量の製品がなければ、不足分は受注残（品切れ）となる。工場の部品pの最大在庫量を$I_{max:fp}$ 、工場、Djおよび各小売店Siにおける製品kの倉庫容量を$J_{max:fk}$, $J_{max:Djk}$, $J_{max:si\ k}$とおく。工場は、簡単のため、各製品の専用ラインを持つものとし、製品kの公称の生産能力をC_{fk}とおく。しかし、機械故障、欠品、欠勤などの確率的変動のためC_{fk} は達成できず、n(=1,2,...)期における生産能力C_{fk} (n)は、各期独立に同一の離散分布に従うものとし、その最小値を$C_{fk:min}$とおく。同様に、各小売店Siの製品kに対するn期の需要量Ds_{ik} (n)も、互いに独立で同一の離散分布に従うものとし、その最小値、最大値、および平均をそれぞれ$D_{min:sik}$、$D_{max:sik}$、Ds_{ik}とおく。満たされなかった需要は受注残となるが、システムの状態数を有限にするため最大$B_{max:sik}$までとし、それを超えた需要は失われるものとする。

ここで、n期首における状態変数として以下を定義する。

$I_{fp}(n)$:工場の部品pの手持ち在庫量

$J_{fk}(n)$:工場の製品kの純在庫量

$B_{Djk}(n)$: Djからの製品kの発注残

$J_{Djk}(n)$: Djの製品kの純在庫量

$B_{Sik}(n)$:小売店Siからの製品kの発注残

$J_{Sik}(n)$:小売店Siの製品kの純在庫量

さらに、n期首に決定する発注、生産、発送量と関連した変数として、

$O_{fp}(n)$:サプライヤーへの部品pの発注量

$P_{fk}(n)$:工場の製品kの生産量

$P'_{fk}(n)$:n期における製品kの実生産量

$O_{Djk}(n)$: Djから工場への製品kの発注量

$O_{Sik}(n)$:小売店SiからDjへの製品kの発注量

$Q_{fp}(n)$: n期首の工場への部品pの発送量

$Q_{Djk}(n)$:工場からDjへの製品kの発送量

$Q_{Sijk}(n)$: Djから小売店Siへの製品kの発送量

を定義し、各期における費用係数として以下を定義する。

C^I_{fp} :工場の部品pの在庫費用/個

C^J_{fk} :工場の製品kの在庫費用/個

C^Q_{fp} :工場への部品pの配送中費用/個

C^B_{fk} :工場の製品kの受注残費用/個

B_{fk} :工場の製品kの受注残発生費用/回

C^J_{Djk} : Djの製品kの在庫費用/個

C^Q_{Djk} : Djへの製品kの配送中在庫費用/個

C^B_{Djk} : Djの製品kの受注残費用/個

B_{Djk} : Djの製品kの受注残発生費用/回

C^J_{Sik} : Siの製品kの在庫費用/個

C^Q_{Sik} : Siへの製品kの配送中在庫費用/個

C^B_{Sik} : Siの製品kの受注残費用/個

B_{Sik} : Siの製品kの受注残発生費用/回

$C_{max:sik}$: Siの製品kにたいして$B_{max:sik}$を超えて失われた需要に対する損失費用/個

なお、純在庫量J_{fk} (n), J_{Djk}(n), Js_{ik} (n)の負の値はそれぞれ工場、Djおよび各小売店Siの受注残を表している。

ここで、n-1期末からn期首における事象の発生順序は次の通りである。

1. 各小売店Siの需要Ds_{ik} (n-1)および工場の生産能力C_{fk} (n-1)が各分布に従い決定される。
2. 工場の実生産量P'_{fk}(n-1)が定まる。

3. 工場、Dj、および各小売店Siへ発送量$Q_{fp}(n\text{-}T_f)$、$Q_{Djk}(n\text{-}T_{Dj})$、$Qs_{ik}(n\text{-}Ts_i)$がそれぞれ到着する。
4. 工場の部品在庫量$I_{fp}(n)$と製品在庫量$J_{fk}(n)$、Djの製品在庫量$J_{Djk}(n)$、発注残情報$B_{Djk}(n)$および各小売店Siの製品在庫量$Js_{ik}(n)$、発注残情報$Bs_{ik}(n)$がそれぞれ決まる。
5. 工場の発注量$O_{fp}(n)$と生産指示量$P_{fk}(n)$、Djの発注量$O_{Djk}(n)$、各小売店Siの発注量$O_{sik}(n)$、およびDjへの発送量$Q_{Dj\,k}(n)$と各小売店Siへの発送量$Qs_{ik}(n)$をそれぞれ決定する。
6. 在庫費用、配送費用、受注残費用、および損失コストが計算される。

第n期首におけるサプライチェーンの状態s_nは、工場における第$(n\text{-}L_f\text{+}T_f\text{+}1)$期から第$(n\text{-}1)$期、Djにおける第$(n\text{-}L_{Dj}\text{+}T_{Dj}\text{+}1)$期から第$(n\text{-}1)$期および各小売店Siにおける$(n\text{-}Ls_i\text{+}Ts_i\text{+}1)$期から第$(n\text{-}1)$期までの部品、製品毎の発注量、工場への第$(n\text{-}T_f\text{+}1)$期から第$n\text{-}1$期、Djにおける第$(n\text{-}T_{Dj}\text{+}1)$期から第$n\text{-}1$期および各小売店Siにおける$(n\text{-}Ts_i\text{+}1)$期から第$n\text{-}1$期までの発送量、および工場の部品在庫量と工場、Dj、各小売店の製品在庫量、発注残情報のベクトルによって表される。これら可能な全ての状態s_nからなる状態空間を$\boldsymbol{S}$とおく。

状態$s_n\in\boldsymbol{S}$における工場の発注量$O_{fp}(n)$、生産指示量$P_{fk}(n)$、Djの製品発注量$O_{Djk}(n)$のとりうる決定の集合は最大在庫量と発注量の制限から各々次式で与えられる。

$$K_{fp}^{O}=\left\{0,\ldots,J_{\max fp}-I_{fp}(n)-\sum_{l=1}^{L_f-T_f-1}O_{fp}(n-l)-\sum_{l=1}^{T_f}Q_{fp}(n-l)\right\} \tag{2.1}$$

$$K_{fk}^{P}=\left\{0,\ldots,\min\left\{I_{fp}(n),C_{fk},J_{\max fk}-J_{fk}(n)\right\}\right\} \tag{2.2}$$

$$K_{Djk}^{O}=\left\{0,\ldots,J_{\max Djk}-\left[J_{Djk}(n)\right]^{+}-\sum_{l=1}^{L_{Dj}-T_{Dj}-1}O_{Djk}(n-l)-B_{Djk}(n)-\sum_{l=1}^{T_{Dj}-1}Q_{Djk}(n-l)\right\} \tag{2.3}$$

ここで、$[x]^{+}=max(0,x)$である。各小売店に対してはその後工程は市場であり、各小売店の可能な製品発注量$O_{sik}(n)$の集合は小売店の倉庫容量、Djの持つ受注残情報、および需要の最小値を用いて次式で与えられる。

$$K_{Sik}^{O}=\left\{0,\ldots,J_{\max Sik}-\left[J_{Sik}(n)\right]^{+}-\sum_{l=1}^{L_{Si}-T_{Si}-1}O_{Sik}(n-l)-B_{Sik}(n)-\sum_{l=1}^{T_{Si}-1}Q_{Sik}(n-l)+D_{\min Sik}\right\} \tag{2.4}$$

工場から各Djへの製品kの可能な発送量の集合は、

$$K_{Dk}^{Q}=\left\{\begin{array}{l}Q_{Djk}(n)\le O_{Djk}(n-L_{Dj}+T_{Dj})+B_{Djk}(n),\\ \sum_{j=1}^{L}Q_{Djk}(n)\le\left[J_{fk}(n)\right]^{+}\text{を満たす}Q_{Djk}(n)\end{array}\right\} \tag{2.5}$$

で与えられる。一方、Djから各小売店Siへの製品kの可能な発送量の集合は、Djへ発注可能な小売店の集合$Si\in Dj$にたいして、

$$K_{Sk}^{Q}=\left\{\begin{array}{l}Q_{Sik}(n)\le O_{Sik}(n-L_{s_i}+T_{s_i})+B_{Sik}(n),\\ \sum_{Si\in Dj}Q_{Sik}(n)\le\left[J_{Djk}(n)\right]^{+}\text{を満たす}Q_{Sik}(n)\end{array}\right\} \tag{2.6}$$

で与えられる。以上から、状態s_nにおける可能な決定

$$\mathbf{a}=(O_f(n),P_f(n),O_{Dj}(n),O_{Si}(n),Q_D(n),Q_S(n)) \tag{2.7}$$

は、$O_f(n)\in K_f^O(s_n)$, $P_f(n)\in K_f^P(s_n)$, $O_{Dj}(n)\in K_{Dj}^O(s_n)$, $O_{Si}(n)\in K_{Si}^O(s_n)$, $Q_D(n)\in K_D^Q(s_n)$, $Q_S(n)\in K_S^Q(s_n)$
を満たさなければならない。与えられる可能な発注量、生産量と発送量の集合の直積を$\boldsymbol{K}(s_n)$で表すことにすれば$\mathbf{a}\in\boldsymbol{K}(s_n)$

である。政策fは、各状態sにおける可能な決定$f(s)$の集合$\{f(s) \in K(s)\ ;\ s \in S\}$である。政策が与えられれば、次期の状態は以下のように定まる。

サプライヤーの供給能力は十分にあるものと仮定しているので、工場へのn期の発送量$Q_{fp}(n)$は、

$$Q_{fp}(n) = O_{fp}(n - L_f + T_f) \tag{2.8}$$

である。一方、Djへのn期の発送量$Q_{Djk}(n)$と小売店Siへのn期の発送量$Q_{Sik}(n)$は、政策として与えられている。このとき、次の期首の状態は以下のように定められる。

$$I_{fp}(n+1) = I_{fp}(n) + Q_{fp}(n - T_f + 1) - P'_{fk}(n) \tag{2.9}$$

$$J_{fk}(n+1) = J_{fk}(n) + P'_{fk}(n) - \sum_{j=1}^{L} O_{Djk}(n - L_{Dj} + T_{Dj} + 1) \tag{2.10}$$

ここで、$P'_{fk}(n)$はn期の実際の生産量であり、

$$P'_{fk}(n) = \min\{P_{fk}(n), C_{fk}(n)\} \tag{2.11}$$

で与えられる。また、

$$J_{Djk}(n+1) = J_{Djk}(n) + Q_{Djk}(n - T_{Dj} + 1) - \sum_{s_i \in Dj} O_{Sik}(n - L_{s_i} + T_{s_i} + 1) \tag{2.12}$$

$$J_{Sik}(n+1) = \max\{J_{Sik}(n) + Q_{Sik}(n - T_{Si} + 1) - D_{Sik}(n), -B_{\max Sik}\} \tag{2.13}$$

である。さらに、Djの受注残$B_{Djk}(n)$と各小売店の受注残$Bs_{ik}(n)$は、各々次式で与えられる。

$$B_{Djk}(n+1) = B_{Djk}(n) + O_{Djk}(n - L_{Dj} + T_{Dj}) - Q_{Djk}(n) \tag{2.14}$$

$$B_{Sik}(n+1) = B_{Sik}(n) + O_{Sik}(n - L_{Si} + T_{Si}) - Q_{Sik}(n) \tag{2.15}$$

これらから、状態s_nで決定$\boldsymbol{a}$をとったとき、次期に状態s_{n+1}へ推移する確率は、生産能力分布および需要分布を用いて次のように与えられる。

$$p(s_n, s_{n+1}, \mathbf{a}) = \begin{cases} \Pr\{C_{fk}(n) = c_{fk}, D_{Sik}(n) = d_{Sik}\},\ s_{n+1} = \\ (O_{fp}(n - L_f + T_f + 2), \ldots O_{fp}(n), Q_{fp}(n - T_f + 1), \ldots Q_{fp}(n), \\ O_{Djk}(n - L_{Dj} + T_{Dj} + 2), \ldots O_{Djk}(n), Q_{Djk}(n - T_{Dj} + 1), \ldots Q_{Dj}(n), \\ O_{Sik}(n - L_{Si} + T_{Si} + 2), \ldots O_{Sik}(n), Q_{Sik}(n - T_{Si} + 1), \ldots Q_{Sik}(n), \ldots \\ I_{fp}(n) + Q_{fp}(n - T_f + 1) - \min\{P_{fk}(n), c_{fk}\}, \\ J_{fk}(n) + \min\{P_{fk}(n), c_{fk}\} - \sum_{j=1}^{L} O_{Djk}(n - L_{Dj} + T_{Dj} + 1), \\ J_{Djk}(n) + Q_{Djk}(n - T_{Dj} + 1) - \sum_{Si \in Dj} O_{Sik}(n - L_{Si} + T_{Si} + 1), \\ \max\{J_{Sik}(n) + Q_{Sik}(n - T_{s_i} + 1) - d_{Sik}, -B_{\max Sik}\}, \\ B_{Djk}(n) + O_{Djk}(n - L_{Dj} + T_{Dj}) - Q_{Djk}(n), \\ B_{Sik}(n) + O_{Sik}(n - L_{Si} + T_{Si}) - Q_{Sik}(n)) \\ 0, \quad otherwise \end{cases} \tag{2.16}$$

さらに状態s_nで決定$\boldsymbol{a}$をとったときのn期における直接費用は次式で与えられる。

$$\begin{aligned} r(s_n, \mathbf{a}) = {} & C^I_{fp} I_{fp}(n) + C^J_{fk}\left[J_{fk}(n)\right]^+ + C^B_{fk}\left[-J_{fk}(n)\right]^+ \\ & + B_{fk} H\left(J_{fk}(n) < 0\right) + C^J_{Djk}\left[J_{Djk}(n)\right]^+ + C^B_{Djk} B_{Djk} \\ & + B_{Djk} H\left(0 > J_{Djk}(n)\right) \\ & + \sum_i \left[C^J_{Sik}\left[J_{Sik}(n)\right]^+ + C_{\max Sik} \sum_{d=D_{\min Sik}}^{D_{\max Sik}} \Pr\{D_{Sik}(n) = d_{sik}\} \right. \\ & \left. \times \left[d_{Sik} - B_{\max Sik} - J_{Sik}(n)\right]^+ + B_{s_i} \Pr(D_{s_i}(n) > J_{s_i}(n)) \right] \\ & + C^Q_f \sum_{l=0}^{T_f - 1} Q_f(n-l) + C^Q_{dc} \sum_{l=0}^{T_{dc} - 1} Q_{dc}(n-l) \\ & + \sum_{s_i \in Shop} C^Q_{s_i} \sum_{l=0}^{T_{s_i} - 1} Q_{s_i}(n-l) \end{aligned} \tag{2.17}$$

ここで$H(e)$は、事象eが起これば値1を、

起こらなければ値0をとる定義関数である。

gを1期当たりの最小平均費用、$h(s_n)$を相対費用とおけば、次の最適性方程式が成り立つ。

$$g+h(s_n)=\min_{\mathbf{a}\in K(s_n)}\{r(s_n,\mathbf{a})+\sum_{s_{n+1}\in \mathbf{S}} p(s_n,s_{n+1},\mathbf{a})h(s_{n+1})\},$$
$$\forall s_n \in \mathbf{S} \qquad (2.18)$$

最適政策f^*は各状態s_nで上式右辺を最小化する決定$f^*(s_n)$として定められる。ここで、相対費用$h(s)$は適当に与えられた状態s_rで$h(s_r)=0$である。

3. 近似DPアルゴリズムSBMPI

以下の状態集合を使用する。

S_T：初期値s_0からはじめて、現在の政策$f(s)$で訪問した状態の集合、訪問回数$v(s)$、$h(s)$を保存。

S_V：これまでの政策で少なくとも一度は訪問したことのある状態の集合、$h(s)$、$f(s)$を保存。

S_U：政策改良の計算で$h(s)$、$f(s)$が未知の状態集合、$h(s)$、$f(s)$を保存。

1. (初期設定)

これまで採用されてきた政策あるいは合理的な政策（例えばSBOSで最適化されたかんばん政策）を初期政策f^0として選択し、日頃よく観測される状態を初期状態s_0として設定する。シミュレーション長m_0, m'を定め、相対値計算の反復回数N_0, N'、停止基準$\varepsilon_1, \varepsilon_2 > 0$、$g(0)=0$、$\beta_1, \beta_2 (0 \le \beta_1, \beta_2 \le 1)$とおき、未知の相対値設定のための非負数$\mu_1, \mu_2$ (μ_1, μ_2 <1)と適当な正数W、初期状態更新状態数γ、最小平均費用の信頼区間推定のための確率αとバッチサイズQ、バッチ数Bを定め、反復回数$k=1$とおく。

2. (Schweitzer変換)

次式を満たす正数τ

$$0<\tau<\min_{\substack{s\in S,\mathbf{a}\in K(s),\\ p(s,s,\mathbf{a})<1}}\{1/(1-p(s,s,\mathbf{a})\}$$

を定め、直接費用$r(s,\mathbf{a})$、推移確率$p(s,s',\mathbf{a})$を以下の式で変換する。

$$r(s,\mathbf{a}) \leftarrow \tau r(s,\mathbf{a}),$$
$$p(s,s',\mathbf{a}) \leftarrow \tau p(s,s',\mathbf{a})+(1-\tau)\delta_{s,s'}$$

ここで$\delta_{s,s'}=1, s=s'; =0, s\neq s'$である。

3. (初期政策のミュレーション)

3-0：集合$S_T=\{s_0\}$、$\#_{old}S_T=1$、訪問回数$q(s_0)=1$、累積費用$TC=0$、$s=s_0$、$m=m_0$、$N=N_0$、$l=0$とおく。

3-1：状態sで初期政策$f(s)=f^0(s)$をとったときの状態推移をシミュレーションし、次期の状態s'を定める。

$TC=TC+r(s,f(s))$とおく。

$s' \notin S_T$ならば$S_T=S_T+\{s'\}$、$q(s')=1$、

$s' \in S_T$ならば$q(s')=q(s')+1$

とおいて、$s=s'$と更新する。

3-2：$l=l+1$として$l \ge m$ならば$\#S_T=|S_T|$とおき、ステップ**3-3**へ。さもなければステップ**3-1**へ。

3-3：$\#S_T > \#_{old}S_T$ならば$\#_{old}S_T = \#S_T$、$m=m+m'$とおき、ステップ**3-1**へ。さもなければステップ**4**へ。

4. （平均費用 g の推定）

平均費用$g(k)$を$g(k)=TC/l$

により計算し、S_Tの中で$q(s)$が最大のsをs_0とおく。

5. (相対値hの推定)

5-1: $s(\neq s_0) \in S_T$に対して

$$h^0(s)=r(s,f(s))-r(s_0,f(s_0))$$

とおき、$h^0(s_0)=0$、$l=0$とおく。

5-2: $s \in S_T$に対して

$$w^{l+1}(s)=r(s,f(s))+\sum_{s'\in S} p(s,s',f(s))h^l(s')$$

を計算する。ここで$p(s,s',f(s))>0$となる$s' \notin S_T$に対しては、**5-2**を通して

$$h^l(s')=r(s',f(s'))-r(s_0,f(s_0))$$

として$w^{l+1}(s)$を計算する。さらに、

$s(\neq s_0) \in S_T$に対して

$$h^{l+1}(s)=w^{l+1}(s)-w^{l+1}(s_0)$$

を計算し$h^{l+1}(s_0)=0$とおく。

5-3: $l=l+1$とおき$l>N$ならばステップ **5-4** へ。さもなければステップ **5-2** へ。

5-4: **(収束判定)**

$s\in S_T$に対して

$$w^{l+1}(s)=r(s,f(s))+\sum_{s'\in S}p(s,s',f(s))h^l(s')$$

を計算する。ここで$p(s,s',f(s))>0$となる$s'\notin S_T$に対しては、

$$h^l(s')=r(s',f(s'))-r(s_0,f(s_0))$$

として$w^{l+1}(s)$を計算する。

$\{s\in S_T\mid q(s)\geq\beta_1 q(s_0)\}$に対して

$$\Delta(s)=\left|w^{l+1}(s)-g(k)-h^l(s)\right|$$

を計算し、$\max_s\Delta(s)>\varepsilon_1$ならば、$N=N+N'$とおき、$s(\neq s_0)\in S_T$に対して

$$h^{l+1}(s)=w^{l+1}(s)-w^{l+1}(s_0)$$

を計算し$h^{l+1}(s_0)=0$、$l=l+1$として、ステップ **5-2** へ。

さもなければ、$s\in S_T$に対して

$w(s)=w^{l+1}(s)$とおいてステップ **6** へ。

6. (SBMPI 初期設定)

$S_V=S_T$とおき $s(\neq s_0)\in S_V$に対して次式を計算し相対値の精度を向上させる。

$$h(s)=w(s)-w(s_0),\quad h(s_0)=0$$

$S_U=\phi$、$s_r=s_0$、$k=k+1$とおく。

7. (政策改良ルーチン)

7-1：$s\in S_V$に対して

$$w(s)=\min_{\mathbf{a}\in N(s,f(s))}\left\{r(s,\mathbf{a})+\sum_{s'\in S}p(s,s',\mathbf{a})h(s')\right\}$$

を計算し、$s(\neq s_0)$に対して

$$h(s)=w(s)-w(s_0),\quad h(s_0)=0$$

とおく。ここで$N(s,f(s))$は$K(s)$における$f(s)$の近傍であり、$p(s,s',\mathbf{a})>0$となる$s'\notin S_V\cup S_U$に対しては、$S_U=S_U\cup\{s'\}$とおき、$f(s')$を初期政策をとる決定と定め、$h(s')=\max\{h(s),\mu_1{}^kW\}$を用いて$w(s)$を計算する。$f(s)$が$w(s)$を与えなければ、$w(s)$を与える任意の決定として$f(s)$を改良する。

7-2：**(収束判定)**

$|g(k)-g(k-1)|<\varepsilon_2$ を満たしかつ$\{s\in S_T\mid v(s)\geq\beta_2 v(s_r)\}$を満たす全ての$s$で$f(s)$が改良されなければ、ステップ **11** へ。さもなければ、**7-3** へ。

7-3：$s\in S_U$に対して

$$w(s)=\min_{\mathbf{a}\in N(s,f(s))}\left\{r(s,\mathbf{a})+\sum_{s'\in S}p(s,s',\mathbf{a})h(s')\right\}$$

を計算し、$h(s)=w(s)-w(s_0)$

とおく。ここで$N(s,f(s))$は$K(s)$における$f(s)$の近傍であり、$p(s,s',\mathbf{a})>0$となる$s'\notin S_V\cup S_U$に対しては

$$h(s')=\max\{h(s),\mu_1{}^kW\}$$

を用いて$w(s)$を計算する。$f(s)$が$w(s)$を与えなければ、$w(s)$を与える任意の決定として$f(s)$を改良する。

8. (シミュレーション)

8-0：$S_T=\{s_0\}$、$\#_{old}S_T$=1、 TC=0、$s=s_0$、$m=m_0$、$N=N_0$、$v(s)=1$、$l=0$とおく。

8-1：状態sで決定$f(s)$をとったときの状態推移をシミュレーションし、次期の状態s'を定め、$TC=TC+r(s,f(s))$、 $s=s'$と更新する。

8-2：$s\notin S_V$かつ$s\notin S_U$ならば、

$S_V=S_V\cup\{s\}$、$S_T=S_T\cup\{s\}$、 $v(s)=1$とおき、$f(s)$を初期政策のとる決定と定め、$h(s)=r(s,f(s))-r(s_0,f(s_0))$とおく。

8-3：$s\notin S_V$かつ$s\in S_U$ならば、

$S_V=S_V\cup\{s\}$、 $S_U=S_U-\{s\}$、 $S_T=S_T\cup\{s\}$、$v(s)=1$とおく。

8-4：$s\in S_V$かつ$s\notin S_T$のとき、$S_T=S_T+\{s\}$、$v(s)=1$とおく。

8-5：$s\in S_V$かつ$s\in S_T$のとき、

$v(s)=v(s)+1$と更新する。

8-6：$l=l+1$として$l\geq m$ならば、$\#S_T=|S_T|$とおき、ステップ **8-7** へ。さもなければステップ **8-1** へ。

8-7：$\#S_T>\#_{old}S_T$ならば$\#_{old}S_T=\#S_T$、$m=m+m'$とおき、ステップ **8-1** へ。さもなければステップ **9** へ。

9. (g の推定)

$g(k)=TC/l$ とおき、S_Tの中で$\nu(s)$が最大のsをs_rとおく。もし $\nu(s_0)\le\gamma$ ならば

$$w(s_0)=r(s_0,f(s_0))+\sum_{s'\in S}p(s_0,s',f(s_0))h(s')$$

$$w(s_r)=r(s_r,f(s_r))+\sum_{s'\in S}p(s_r,s',f(s_r))h(s')$$

を計算する。ここで $p(s_r,s',f(s))>0$ となる $s'\notin S_V\cup S_U$ に対しては、初期政策 $f(s')$ を用いた

$$h\ (s')=r(s',f(s'))-r(s_0,f(s_0))$$

としてwを計算する。$s(\neq s_r)\in S_V\cup S_U$ にたいして、$h(s)=h(s)+w(s_0)-w(s_r)$ とおき、$h(s_r)=0$、$s_0=s_r$とおく。

10. (h(s)の推定)

10-1: $s(\neq s_0)\in S_V\cup S_U$ に対して

$$h^0(s)=h(s)$$

とおき、$h^0(s_0)=0$、$l=0$とおく。

10-2: $s\in S_T$に対して

$$w^{l+1}(s)=r(s,f(s))+\sum_{s'\in S}p(s,s',f(s))h^l(s')$$

を計算する。ここで **10-2** を通して $p(s,s',f(s))>0$ となる $s'\notin S_V\cup S_U$に対しては、初期政策$f(s')$を用いた

$$h^l(s')=r(s',f(s'))-r(s_0,f(s_0))$$

として$w^{l+1}(s)$を計算する。さらに、$s(\neq s_0)\in S_T$に対して

$$h^{l+1}(s)=w^{l+1}(s)-w^{l+1}(s_0)$$

を計算し$h^{l+1}(s_0)$=0とおく。

10-3: $l=l+1$とおき$l>N$ならばステップ **10-4** へ。さもなければステップ **10-2** へ。

10-4: **(収束判定)**

$s\in S_T$に対して

$$w^{l+1}(s)=r(s,f(s))+\sum_{s'\in S}p(s,s',f(s))h^l(s')$$

を計算する。ここで $p(s,s',f(s))>0$ となる $s'\notin S_V\cup S_U$に対しては、初期政策 $f(s')$ を用いた

$$h^l(s')=r(s',f(s'))-r(s_0,f(s_0))$$

として$w^{l+1}(s)$を計算する。

$\{s\in S_T\mid \nu(s)\ge\beta_1\nu(s_r)\}$に対して

$$\Delta(s)=\left|w^{l+1}(s)-g(k)-h^l(s)\right|$$

を計算し、$\max_s\Delta(s)>\mu_2\varepsilon_1$ ならば、$N=N+N'$とおき、$s(\neq s_0)\in S_T$に対して $h^{l+1}(s)=w^{l+1}(s)-w^{l+1}(s_0)$ を計算し$h^{l+1}(s_0)$=0、$l=l+1$として、ステップ **10-2** へ。さもなければ、$s(\neq s_0)\in S_V$に対して $h(s)=w^{l+1}(s)-w^{l+1}(s_0)$、 $h(s_0)=0$ とおき、$k=k+1$ とおいてステップ **7** へ。

11. (最小平均費用g^*の区間推定)

11-1：求める準最適政策と相対値は、$\{f(s),h(s);s\in S_T\}$で与えられる。以下、最小平均費用g^*の 100(1-α)%信頼区間を推定する。$TC=0$、$b=1$、$l=1$とおく。$s=s_0$とおく。

11-2：状態sで決定$f(s)$をとったときの直接費用を計算し、状態推移をシミュレーションして次期の状態s'を定める。$TC=TC+r(s,f(s))$、 $s=s'$と更新する。もし$s\notin S_V\cup S_U$ならば、$f(s)$を初期政策に取る。

11-3：$l=l+1$とおき、$l>Q$ならば **11-4** へ。さもなければ、**11-2** へ。

11-4：$g(b)=TC/Q$、$b=b+1$ とおき、$b>B$ ならばステップ **11-5** へ。さもなければ $TC=0$、$l=1$、$s=s_0$とおき、ステップ **11-2** へ。

11-5：$\{g(1)/\tau,...,g(B)/\tau\}$の

平均 $g=\sum_{i=1}^{B}(g(i)/\tau)/B$

と分散 $S^2=\sum_{i=1}^{B}(g(i)/\tau-g)^2/(B-1)$ を計算し、自由度$(B-1)$のt分布の両側α点の値を$t_\alpha(B-1)$としたとき、最小平均費用g^*の$100(1-\alpha)$%信頼区間は

$$[g-t_\alpha(B-1)S/\sqrt{B},g+t_\alpha(B-1)S/\sqrt{B}]$$

で与えられる。

4. 今後の課題

2節のUMDPを解く3節の近似DPアルゴリズムSBMPIのプログラムを作成する

のが第1の今後の課題である。

これまでに、手持ちの製品在庫を複数の下流拠点へ最適に配分する割当問題が、様々な仮定のもとで理論的に研究されている。第2の課題は、これらの結果を、近似DPアルゴリズムSBMPIにより計算された準最適発注・発送政策と比較することで、近似の妥当性を判定することができる。さらに、多品種SCへのプル方式の適用を上の結果を踏まえて定式化し、現実規模のSCへ各プル方式を適用できるように、SBOSアルゴリズムを改良する。そして、各プル方式のパラメータを最適設定した性能を求め、準最適発注・生産・発送政策を基準とした各プル方式の性能比較を行う。さらに、かんばん方式では**引き取りかんばん**が発注量のみならず、「**先入れ先出し（先着順）**」の原則の下、発送量をも決定している。「先入れ先出し」を陽に扱うためには、小売店における各品種の需要の発生時刻と配送センターや工場からの各品種の発送時刻および工場における各品種の生産完了時刻を政策決定時点とする時間平均セミ・マルコフ決定過程(undiscounted semi-Markov decision process, **USMDP**)として定式化しなければならない。USMDPに対する近似DPアルゴリズムをSBMPIアルゴリズムに倣って開発し、準最適発注・生産・発送政策を計算することで、これまで何の疑いもなく採用されてきた「先入れ先出し」がどの程度実際に有効か、あるいは準最適発送政策によらざるをえないかを知ることができる。

参考文献

[1] T.G. de Kok and J.C. Fransoo （大野訳）、“サプライチェーンの運用計画：計画概念の定義と比較”、pp. 559-631、黒田充、大野勝久監訳「サプライチェーンハンドブック」第12章、朝倉書店 (2008)

[2] 大野勝久、「サプライチェーンの最適運用－かんばん方式を超えて」、朝倉書店 (2011)

[3] K. Ohno, "The optimal control of just-in-time-based production and distribution systems and performance comparisons with optimized pull systems," *European Journal of Operational Research,* Vol.213, pp. 124-133 (2011)

[4] R. Bellman, *Dynamic Programming*, Princeton Univ. Press (1957)

[5] K. Ohno and K. Ichiki, "Computing optimal policies for controlled tandem queueing systems," *Operations Res.*, Vol. 35, pp.121-126 (1987)

[6] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*, Athena Scientific (1996)

[7] J. Si, A. Barto, W. Powell and D. Wunsch ed., *Handbook of Learning and Approximate Dynamic Programming*, IEEE Press (2004)

[8] W. B. Powell, *Approximate Dynamic Programming – Solving the Curses of Dimensionality*, Wiley-Interscience (2007)

[9] T.K. Das, A. Gosavi, S. Mahadevan and N. Marchalleck, "Solving semi-Markov decision problem using average reward reinforcement learning", *Management Science*, Vol. 45, No.4, pp. 560-574(1999)

[10] A. Gosavi, N. Bandla and T. K. Das, "A reinforcement learning approach to a single leg airline revenue management problem with multiple fare classes and overbooking", *IIE Transactions*, Vol. 34, pp. 729-742 (2002)

[11] Y. He, M.C. Fu and S.I. Marcus, "A simulation-based policy iteration algorithm for average cost unichain Markov decision processes", M. Laguna and J.L. G. Velarde eds., *Computing Tools for Modeling, Optimization and Simulation*, pp. 161-182, Kluwer Academic (2000)

[12] 大野, 八嶋, 伊藤、"ニューロ・ダイナミックプログラミングによる生産ラインの最適制御に関する研究", 日

本経営工学会論文誌, Vol. 54, No. 5, pp. 316-325 (2003)
[13] 大野, 伊藤、”ニューロ・ダイナミックプログラミングによる生産･物流システムの最適制御とプル方式の比較”, 日本経営工学会論文誌, Vol.55, No.4, pp.174-188 (2004)
[14] 大野、”生産･物流システムの最適制御と JIT”, ジャストインタイム生産システム研究会編「ジャストインタイム生産システム」, pp.351-377, 日刊工業新聞社 (2004)
[15] 大野、“プル方式の最適化”, pp.657-664, 大野、岡本他編集 「進化技術ハンドブック 第Ⅲ巻 応用編：生産・物流システム」、第 30.1 節、電気学会 進化技術応用専門委員会編、近代科学社 (2012)
[16] 伊藤, 田村, 大野: “プル方式の最適化”, pp.664-670, 第 30.2 節、同上.
[17] 大野, 坊, 田村：日本 OR 学会秋季研究発表会アブストラクト集, pp.118-119 (2012)
[18] K. Ohno, T. Boh and T. Tamura: “New Approximate Dynamic Programming Algorithms for Large-scale Undiscounted Markov Decision Processes: Performance Comparisons with Optimized Pull Systems” 投稿中(2013)

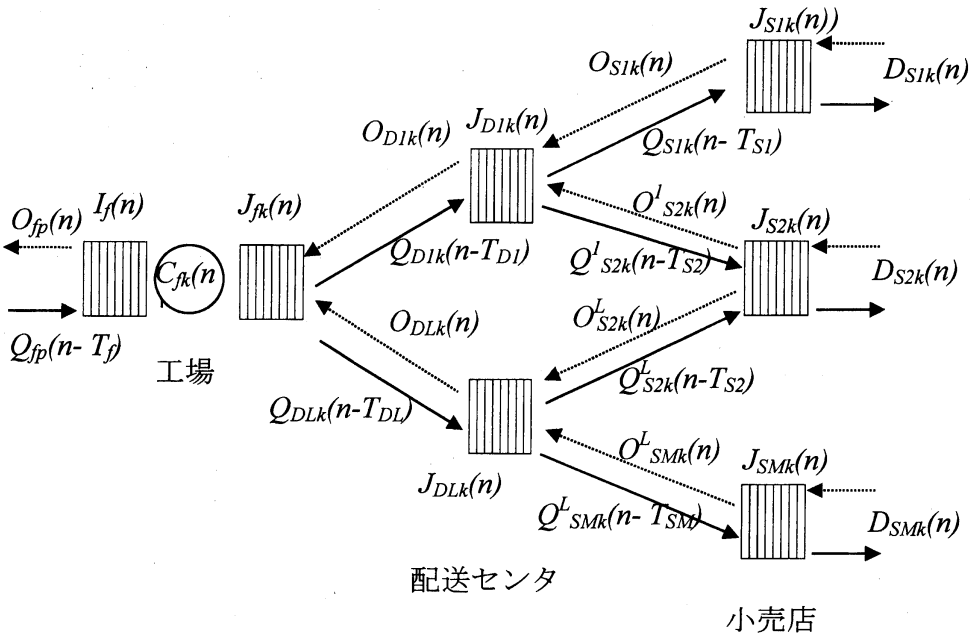

図 1. サプライチェーンシステム